

14
CLAIMS

What is claimed is:

1. A method for serializing objects in a compiled programming language, comprising the steps of:
 - 5 creating a storage agent for a serializable object that is configured to construct instances of the serializable object;
 - registering a serializable object name and an associated storage agent pointer in a type map for each serializable object, wherein the storage agent pointer links to the corresponding storage agent;
 - 10 storing the serializable object with a serializable object name and object data on an electronic storage medium;
 - identifying the serializable object's storage agent by using the serializable object name to index into the type map when the serialized object is read from the electronic storage medium;
 - 15 constructing a serializable object instance using the identified storage agent; and loading the serializable object instance with the object data read from the electronic storage medium to restore the serialized object.
2. A method as in claim 1, wherein the step of identifying the serializable object's storage 20 agent further comprises the step of matching the serializable object name read back from the electronic storage medium with the storage agent pointer of a serializable object using the type map.
- 25 3. A method as in claim 2, further comprising the step of recreating an object type with the storage agent using a template generated method to create an instance of the serializable object.
4. A method as in claim 1; wherein the step of registering the serializable object in a type map further comprises the step of storing the serializable object names and their 30 associated storage agent pointers in a custom map data structure.
5. A method as in claim 1, wherein the step of registering the serializable object in a type map further comprises the step of using a dynamic vector as the type map.

6. A method as in claim 1, wherein the step of registering a serializable object name further comprises the step of storing serializable object names in the type map that are set by an application programmer using the source code.
- 5 7. A method as in claim 1, wherein the step of creating a storage agent for a serializable object further comprises the step of creating a storage agent for a serializable class.
8. A method as in claim 1, wherein the step of creating a storage agent for a serializable object further comprises the step of creating a static storage agent for a serializable class.
- 10 9. A method as in claim 1, wherein the step of creating a storage agent for a serializable object further comprises the step of creating a storage agent having an object factory for creating instances of the serializable object.
- 15 10. A method for recreating a serializable object in a compiled programming language, comprising the steps of:
 - reading a serializable object from an electronic data stream where the serializable object has been stored;
 - receiving an object name with the serializable object;
 - 20 finding a named storage agent for the serializable object based on the object name received for the serializable object;
 - creating a new instance of the serializable object using the named storage agent;
 - calling a restore method to load serialized data for the serializable object from the electronic data stream into the new instance of the serializable object.
- 25 11. A method as in claim 10, wherein the step of creating a new instance of the serializable object using the named storage agent further comprises the step of creating a new instance of the serializable object to restore an object stored by reference.
- 30 12. A method as in claim 10, further comprising the step of creating a named storage agent for the serialized object based on an object name of the serialized object.

13. A method as in claim 12, further comprising the step of identifying serializable object names set by an application programmer at compile time and creating named storage agents for the serializable object names.

5 14. A method as in claim 10, wherein the step of finding a named storage agent for the serializable object further comprises the step of finding a named storage agent using a type map which stores the object name associated with a pointer to the corresponding named storage agent.

10 15. A method as in claim 13, further comprising the step of accessing the pointer to the named storage agent to call a function that creates a serializable object corresponding to the object name.

15 16. A method as in claim 10, wherein the step of finding a named storage agent for a serializable object further comprises the step of finding a named storage agent for a serializable class.

20 17. A method as in claim 10, further comprising the step of creating a storage agent for a serializable object that is static.

25 18. A system for serializing objects in a compiled programming language, comprising:
 a mapping object configured to store serializable object names and corresponding named storage agent pointers;
 a named storage agent, referenced by the named storage agent pointer, being configured to create an instance of a serializable object when the serializable object is restored;
 a parent storage agent object configured to operate on the mapping object and to provide virtual functions for the named storage agent; and
 an archive object in communication with the parent storage agent and the named storage agent, the archive object being configured to save and restore the serialized object.

30

19. A system as in claim 18, wherein the mapping object enables the archive object to create new instances of serializable objects using the named storage agent and to restore an object's state into the new instances of serializable objects.

5 20. A system as in claim 18, wherein the archive object further comprises an output stream object to write the serializable object to an output stream.

21. A system as in claim 18, wherein the archive object further comprises an input stream object to read the serializable object from an input stream.

10 22. A system as in claim 18, wherein the serializable object names are entered into the mapping object as set by an application programmer at compile time.

15 23. A system as in claim 18, wherein a serialized object further comprises an archive header including an application name, application major revision, and an application minor revision.

24. A system as in claim 18, wherein a serialized object further comprises a storage mode flag to indicate whether the serialized object is stored by value or stored by reference.

20 25. A system as in claim 18, wherein the serialized object further comprises run time type information that includes the serialized object's class name.

25 26. A system as in claim 18, further comprising an electronic file within which a plurality of serialized objects are stored.

27. A system for serializing objects in a compiled programming language, comprising:

- a mapping object means for storing serializable object names and corresponding named storage agent pointers;
- 5 a named storage agent means, referenced by the named storage agent pointer, for creating an instance of a serializable object when the serializable object is restored;
- a parent storage agent means for operating on the mapping object means and to provide virtual functions for the named storage agent means;
- 10 an archive object means in communication with the parent storage agent means and the named storage agent means, wherein the archive object means is for saving and restoring the serialized object; and
- wherein the mapping object means enables the archive object means to create new instances of serializable objects using the named storage agent means and to restore an object's state into the new instances of serializable objects.